Abstract:
Composite Galois Field $GF((2^m)^n)$ multiplications denote the multiplication with extension field over the ground field $GF(2^m)$, that are used in cryptography and error correcting codes. In this paper, composite versatile and vector $GF((2^m)^2)$ multipliers are proposed. The proposed versatile $GF((2^m)^2)$ multiplier design is used to perform the $GF((2^x)^2)$ multiplication, where $2 \leq x \leq m$. The proposed vector $GF((2^m)^2)$ multiplier design is used to perform $2^k$ numbers of $GF((2^{2x})^2)$ multiplications in parallel, where throughput is comparatively higher than other designs and $k \in \{0, 1, \ldots (log_2 m) - 1\}$. In both the works, the hardware cost is the trade-off while the flexibility is high. The proposed and existing multipliers are synthesised and compared using 45 nm CMOS technology. The throughputs of the proposed parallel and serial vector $GF((2^x)^2)$ multipliers are 72.7% and 53.62% greater than Karatsuba based multiplier design [11] respectively.

Mohamed Asan Basiri M and Sandeep K Shukla, Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, Utter Pradesh - 208016, India
1 Introduction

The hardware based Galois field multipliers play major role in crypto graphy [1] and error correcting codes. The parameter metrics involved in any hardware design are throughput and hardware cost (area and power dissipation). Here, anyone can be optimized at the cost of others. The papers [2], [3], and [4] show Reed Solomon code related applications, which require \(GF(2^m)\) multipliers for various \(m\) values. Vector processing is an important feature in hardware designs, where more number of similar operations can be done in parallel. Homomorphic encryption using matrices is explained in [5], where vector Galois field multiplications are performed using GPU.

1.1 Related Works

The following literatures are found in composite Galois field multiplications. The papers [6] [7] [8] [9] explain the composite Galois field \(GF((2^m)^n)\) multiplication. The equations from (1) to (9) show the \(GF((2^m)^n)\) multiplication. Here, \(A\) and \(B\) are the inputs while the output is \(Y\). The irreducible polynomials over \(GF(2^n)\) and \(GF(2^m)\) are \(Q\) and \(P\) respectively.

\[
Y = A.B \mod Q = Y_{n-1}x^{n-1} + Y_{n-2}x^{n-2} + ... Y_0
\]  

(1)

\[
A = (A_{n-1}x^{n-1} + A_{n-2}x^{n-2} + ... A_0), \ A_i \in GF(2^m)
\]  

(2)

\[
B = (B_{n-1}x^{n-1} + B_{n-2}x^{n-2} + ... B_0), \ B_i \in GF(2^m)
\]  

(3)

\[
Q = (x^n + Q_{n-1}x^{n-1} + ... + Q_1x + Q_0), \ Q_i \in GF(2^m)
\]  

(4)

\[
A_i = (a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + ... a_0), \ a_i \in GF(2)
\]  

(5)

\[
B_i = (b_{m-1}x^{m-1} + b_{m-2}x^{m-2} + ... b_0), \ b_i \in GF(2)
\]  

(6)

\[
Y = \{ \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} X_{ij}x^{i+j} \} \mod (Q_{n-1}x^{n-1} + ... + Q_1x + Q_0)
\]  

(7)

\[
X_{ij} = A_i.B_j \mod P, \ X_{ij} \in GF(2^m); \ 0 \leq i, j \leq (n-1)
\]  

(8)

\[
P = (x^n + p_{m-1}x^{n-1} + ... p_0), \ p_i \in GF(2)
\]  

(9)

The aforementioned equations can be reduced for \(GF((2^m)^2)\) as shown in (10) to (14), which requires six \(GF(2^m)\) multiplications. Here, \(A_1, A_0, B_1, B_0, Q_1, Q_0 \in GF(2^m)\).

\[
Y = Y_1x + Y_0 = (A_1x + A_0).B_1x + B_0
\]  

(10)

\[
Y = A_1.B_1x^2 + (A_1.B_0 + B_1.A_0)x + A_0.B_0
\]  

(11)

\[
Y = A_1.B_1.(Q_1x + Q_0) + (A_1.B_0 + B_1.A_0)x + A_0.B_0
\]  

(12)

\[
Y_1 = (A_1.B_0 + B_1.A_0 + A_1.B_1.Q_1)
\]  

(13)

\[
Y_0 = (A_0.B_0 + A_1.B_1.Q_0)
\]  

(14)

Hybrid field \(GF((2^m)^n)\) multiplier is shown in (10) that requires \(n\) cycles. Also, the multiplier circuit proposed in [10] is used for \(GF(2^m)\) multiplication if \(n = 1\). The Karatsuba based \(GF((2^m)^2)\) multiplication is shown in [11], where the composite irreducible polynomial is considered as \(Q = (x^2 + x + Q_0)\). Hence, the aforementioned equations can be changed into as shown in (15) to (16), which requires four multiplications.

\[
L = A_0.B_0; \ H = A_1.B_1; \ K = (A_1 + A_0).B_1 + B_0
\]  

(15)

\[
Y_1 = K \oplus L; \ Y_0 = (Q_0.H) \oplus L
\]  

(16)
1.2 Contribution of This Paper

In this paper, composite versatile and vector $GF((2^m)^2)$ multipliers are proposed. The proposed versatile $GF((2^m)^2)$ multiplier design is used to perform the $GF((2^x)^2)$ multiplication, where $2 \leq x \leq m$. Here, the term versatility is to mention the ability of $GF((2^m)^2)$ multiplier to perform various length multiplications ($GF((2^x)^2)$ multiplications with $2 \leq x \leq m$), where one multiplication can be done at a time. The proposed vector $GF((2^m)^2)$ multiplier design is used to perform $2^k$ numbers of $GF((2^m)^2)$ multiplications in parallel, where $k \in \{0, 1, \ldots, (\log_2 m) - 1\}$. Here, the term vector is to mention the ability of $GF((2^m)^2)$ multiplier to perform multiple multiplications at a time. In both the works, the hardware cost is the trade-off while the flexibility is high. Also, the objective of the proposed vector designs is to improve the throughput. The proposed and existing multipliers are synthesised and compared using 45 nm CMOS technology.

The rest of the paper is organized as follows: Section 2 states the proposed versatile and vector $GF((2^m)^2)$ multipliers. Design modelling, implementation, and results are shown in Section 3, followed by a conclusion in Section 4.

Here, $A_0$, $A_1$, $B_0$, $B_1$, $Q_0$, $Q_1$, and $s$ are input ports; $Y_0$ and $Y_1$ are output ports.

Figure 1: $GF((2^8)^2)$ proposed (a) parallel (b) serial versatile/vector multipliers designs

2 Proposed versatile and vector $GF((2^m)^2)$ multipliers

Fig. 1(a) and (b) show the $GF((2^8)^2)$ proposed parallel and serial versatile/vector multipliers respectively. According to the equations (13) and (14), the 8-bits inputs are $A_1$, $A_0$, $B_1$, $B_0$, $Q_1$, and $Q_0$. Here, the interconnect circuit is used for versatile and vector compatibility. The control line $s$ is used to perform various vector and versatile multiplications using the proposed architecture, which are shown in Table 2. As shown in Fig. 1(a) and (b), the critical path includes a multiplier, interconnect unit, and two 2-bit XOR gates. The critical path depth of Fig. 1(a) and (b) are shown in equation (17). Here, $T(XOR)$, $T(MUL)$, and $T(CON)$ are circuit depth of 2-bit XOR gate, proposed $GF(2^m)$ multiplier, and proposed interconnect circuit respectively. The proposed parallel and serial $GF((2^m)^2)$ multipliers require one and two cycles to produce the output. Table 1 shows

Here, $a$ and $b$ are input ports; $o_2$, $o_3$, $o_4$, $o_5$, $o_6$, $o_7$, and $o_8$ are output ports.

Figure 2: $GF(2^8)$) proposed (a) versatile multiplier with cell architectures (b) first PE, (c) inter PE, and (d) last PE
Here, \(a\) and \(b\) are input ports; \(o_{20}, o_{21}, o_{22}, o_{23}, o_{40}, o_{41}\), and \(o_8\) are output ports.

Figure 3: \(GF(2^8)\) proposed (a) vector multiplier with cell architectures (b) inter PE-S0, (c) inter PE-S1, and (d) inter PE-S2

Here, \(i\), irreducible polynomial \(p\), and control line \(s\) are inputs; \(o\) is the output port.

Figure 4: Unit reduction circuit (URC) for \(GF((2^8)^2)\) proposed (a) vector and (b) versatile multipliers

the operation of \(GF((2^8)^2)\) proposed serial versatile/vector multipliers. The multipliers used in
In (a), $o_2$, $o_3$, $o_4$, $o_5$, $o_6$, $o_7$, and control line $s$ are input ports while the output port is $out$; In (b), $o_{20}$, $o_{21}$, $o_{22}$, $o_{23}$, $o_{40}$, $o_{41}$, $o_{8}$, and control line $s$ are input ports while the output port is $z$.

Figure 5: Interconnect circuit for $GF(2^8)^2$ proposed (a) versatile, (b) vector multipliers, and (c) AND gate with $(m + 1)$-bit input $z$ and 1-bit input $y$.

Fig. 1(a) and (b) follow the algorithm as shown in 1. Accordingly, the multiplier architectures are proposed in the Fig. 2 and Fig. 3 to perform versatile and vector operations respectively.

Algorithm 1 Algorithm for $GF(2^m)$ multiplication

Input: $A(x)$, $B(x)$, and irreducible polynomial $G(x) = x^m + P(x)$, where $A(x)$, $B(x)$, and $P(x)$ are of degree $(m - 1)$.

Output: $C(x)^m = A(x).B(x) \mod G(x)$, where the degree of the output is $(m - 1)$

1: $C^0(x) = 0$ and $A^0(x) = A(x)$
2: for $i = 1$ to $m$ do
3:  $C^i(x) = b_{i-1}A^{i-1}(x) + C^{i-1}(x)$
4:  $A^i(x) = (A^{i-1}(x) \ll 1) \mod G(x)$  // unit reduction step
5: end for

$$T_{\text{critical}} = T(MUL) + T(CON) + 2T(XOR)$$ (17)
$$T(\text{verse}, m) = T(\text{first PE}) + T(\text{last PE}) + T(\text{inter, verse})$$ (18)
$$T(\text{vector, m}) = T(\text{first PE}) + T(\text{last PE}) + T(\text{inter, vector})$$ (19)

Algorithm 2 Proposed algorithm for vector $GF((2^x)^2)$ multiplication using $GF((2^m)^2)$ multiplier, where $x = \frac{m}{k}$, $k \in \{0, 1, ..., (log2m) - 1\}$

Input: $A_0$, $A_1$, $B_0$, $B_1$, $Q_0$, $Q_1$ with $m$-bits wide

Output: $Y_0^c$ and $Y_1^c$ with $m$-bits wide, where $c = 0$ to $(2^k) - 1$.

1: Parallel for $c$ to $(2^k) - 1$ do
2:  $[\pi] = (((c + 1)x - 1), ..., (cx + 1), cx)$, where $msb$ and $lsb$ are $(c + 1)x - 1$ and $cx$ respectively.
3:  $Y_0^c[\pi] = ((A_0^c[\pi].B_0^c[\pi]) + (B_1^c[\pi].A_0^c[\pi])) + (A_1^c[\pi].B_0^c[\pi]Q_1[\pi])$;
4:  $Y_1^c[\pi] = ((A_0^c[\pi].B_0^c[\pi]) + (A_1^c[\pi].B_1^c[\pi]Q_0[\pi])$;
5: end for

Algorithm 2 shows the proposed vector $GF((2^x)^2)$ multiplication using $GF((2^m)^2)$ multiplier, where $x = \frac{m}{k}$, $k \in \{0, 1, ..., (log2m) - 1\}$, $2^k$ is the total number of vector operations possible, and $[\pi]$ represents the bits involved in the each vector operation. In 2-bit vector multiplication using
Table 1: Operation of $GF((2^8)^2)$ proposed serial versatile/vector multipliers as shown in Fig. 1(b)

<table>
<thead>
<tr>
<th>Cycle</th>
<th>$Y_0$</th>
<th>$Y_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$B_1.A_0$</td>
<td>$B_1.A_1$</td>
</tr>
<tr>
<td>2</td>
<td>$B_1.A_1.Q_0+B_0.A_0$</td>
<td>$B_1.A_0+B_1.A_1.Q_1+B_0.A_1$</td>
</tr>
</tbody>
</table>

$GF((2^8)^2)$, $[\pi]$ will be $[1,0]$, $[3,2]$, $[5,4]$, and $[7,6]$, where $0 \leq c \leq 3$. In 4-bit vector multiplication using $GF((2^8)^2)$, $[\pi]$ will be $[3,2,1,0]$ and $[7,6,5,4]$, where $0 \leq c \leq 1$. In 8-bit multiplication using $GF((2^8)^2)$, $[\pi]$ will be $[7,6,5,4,3,2,1,0]$, where $c = 0$. Fig. 2(a) shows the $GF(2^8)$ proposed versatile multiplier, whose first, inter, and last processing elements (PEs) are shown in Fig. 2(b), (c), and (d) respectively. Fig. 3(a) shows the $GF(2^8)$ proposed vector multiplier, whose first PE, last PE, inter PE, inter PE-S0, inter PE-S1, and inter PE-S2 are shown in Fig. 2(b), Fig. 2(c), Fig. 2(d), Fig. 3(b), Fig. 3(c), and Fig. 3(d) respectively.

\[
T(\text{inter, verse}) = (m - 2).T(\text{inter PE}) \tag{20}
\]

\[
T(\text{inter, vector}) = T(\text{inter}) + T(\text{inter}_S) \tag{21}
\]

\[
T(\text{inter}_S) = (\log_2 m)(\text{inter PE}_S) \tag{22}
\]

\[
T(\text{inter}) = (m - (\log_2 m) - 2).T(\text{inter PE}) \tag{23}
\]

\[
T(\text{first PE}) = T(\text{URC}) \tag{24}
\]

\[
T(\text{last PE}) = T(\text{AND}) + T(\text{XOR}) \tag{25}
\]

\[
T(\text{inter PE}) = T(\text{URC}) \tag{26}
\]

\[
T(\text{inter PE}_S) = T(\text{URC}) + T(\text{MUX}) \tag{27}
\]

\[
T(\text{arc, verse}) = T(\text{TRI}) + T(\text{AND}) + T(\text{XOR}) \tag{28}
\]

\[
T(\text{arc, vector}) = T(\text{AND}) + T(\text{arc, M}) + T(\text{XOR}) \tag{29}
\]

\[
T(\text{arc, M}) = ((\log_2 m) - 1).T(\text{MUX}) \tag{30}
\]

Table 2: Proposed composite versatile and vector $GF((2^8)^2)$ multiplications with respect to control line

<table>
<thead>
<tr>
<th>Control line $(s[7:0])$</th>
<th>Versatile multiplication</th>
<th>Control line $(s[2:0])$</th>
<th>Vector multiplications</th>
</tr>
</thead>
<tbody>
<tr>
<td>000000010</td>
<td>one $GF((2^8)^2)$</td>
<td>000</td>
<td>four $GF((2^8)^2)$</td>
</tr>
<tr>
<td>000000100</td>
<td>one $GF((2^8)^2)$</td>
<td>101</td>
<td>two $GF((2^8)^2)$</td>
</tr>
<tr>
<td>000001000</td>
<td>one $GF((2^8)^2)$</td>
<td>100</td>
<td>two $GF((2^8)^2)$ and one $GF((2^8)^2)$</td>
</tr>
<tr>
<td>000100000</td>
<td>one $GF((2^8)^2)$</td>
<td>001</td>
<td>two $GF((2^8)^2)$ and one $GF((2^8)^2)$</td>
</tr>
<tr>
<td>001000000</td>
<td>one $GF((2^8)^2)$</td>
<td>111</td>
<td>one $GF((2^8)^2)$</td>
</tr>
<tr>
<td>010000000</td>
<td>one $GF((2^8)^2)$</td>
<td>others</td>
<td>undefined</td>
</tr>
<tr>
<td>100000000</td>
<td>one $GF((2^8)^2)$</td>
<td>undefined</td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>undefined</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The critical path of $GF(2^m)$ proposed multiplier block used in $GF((2^m)^2)$ versatile multiplier is shown in equation (18). Here, $T(\text{first PE})$, $T(\text{last PE})$, and $T(\text{inter PE})$ are critical path depth of first, last, and inter PEs used in proposed versatile/vector $GF(2^m)$ multipliers respectively. The critical path of $GF(2^m)$ proposed multiplier block used in $GF((2^m)^2)$ vector multiplier is shown in equation (19). Here, $T(\text{inter}_S)$ represents the combined critical path of inter PE-S0, inter PE-S1, and inter PE-S2. Similarly, $T(\text{URC})$, $T(\text{AND})$, $T(\text{XOR})$, and $T(\text{MUX})$ represent the critical path depth of unit reduction circuit (URC), 2-bit AND gate, 2-bit XOR gate, and 2-to-1 multiplexer respectively. The critical path of URC used in proposed versatile $GF(2^m)$ multiplier as shown in Fig.4(a) includes a tri-state buffer, 2-bit AND gate and 2-bit XOR gate. Similarly, the critical path of URC used in proposed vector $GF(2^m)$ multiplier as shown in Fig.4(a) includes a three 2-to-1 multiplexers, two 2-bit AND gate and 2-bit XOR gate. The critical path depth of URC used in proposed versatile and vector $GF(2^m)$ multiplier multipliers are shown in equations (28) and (29) respectively. Here, $T(\text{TRI})$ represents the critical path of tri-state buffer. Fig. 4(a) and (b) represent URC for $GF((2^8)^2)$ proposed vector and versatile multipliers respectively. Fig. 5(a) and (b) represent interconnect circuits for $GF((2^8)^2)$ proposed versatile and vector multipliers respectively. The critical paths of interconnect circuits (ICs) for $GF((2^8)^2)$ proposed versatile and
vector multipliers are shown in the equations (31) and (32) respectively. Fig. 5(c) represents the AND gate used in Figs. 2, 3, and 5. Here, one bit input (y) is bit wise ANDed with (m + 1)-bit input (z) to produce (m + 1)-bit output. In overall, \( GF(2^m)^2 \) proposed versatile/vector parallel and serial multipliers are designed using six and four \( GF(2^m) \) multipliers, that are connected to dedicated interconnect circuits (CONs). The \( GF(2^m) \) multipliers/CONs are used to perform the required multiplication in versatile and vector \( GF((2^m)^2) \) multipliers with respect to the m-bit and \( \log_2 m \)-bit control lines (s) respectively.

\[
T(\text{CON, verse}) = T(\text{AND}) + (\log_2 (m - 1))T(\text{XOR}) \tag{31}
\]

\[
T(\text{CON, vector}) = ((\log_2 m) - 1))T(\text{MUX}) \tag{32}
\]

3 Design modelling, Implementation, and Results

Table 3 shows a theoretical comparison of various composite Galois field \( GF((2^m)^2) \) multiplier designs. Here, the versatility for proposed vector multiplier designs are marginal because these can be used to perform \( GF((2^2)^2) \) multiplications, where \( x = \frac{m}{2} \), \( k \in \{0, 1, \ldots, \log_2 m\} \}. \) For example, proposed vector \( GF((2^8)^2) \) can be used to perform \( GF((2^2)^2), GF((2^4)^2), \) and \( GF((2^8)^2) \) multiplications only. Here, \( GF((3^2)^2), GF((5^2)^2), GF((7^2)^2) \) cannot be performed, whereas in proposed versatile design, these multiplications are possible. Table 4 shows the comparison of critical path delay, frequency, area, switching power, leakage power, power delay product (PDP), and throughput between various \( GF((2^m)^2) \) multiplier designs using 45 nm CMOS library with Cadence Genus and Innovus. Here, the proposed parallel designs require more area than proposed serial designs. On the other hand, the proposed serial designs require more cycles than the proposed parallel designs. Since the hybrid multiplier design [10] requires only one multiplier in the critical path, the delay for [10] is less than others. Also, the proposed designs include one proposed multiplier along with the interconnect circuit in their critical path. Therefore, the delay for proposed designs are greater than [10]. The throughput of the proposed vector designs is much higher than other designs. The throughputs of the proposed parallel and serial vector \( GF((2^8)^2) \) multipliers are 72.7% and 53.62% greater than Karatsuba based multiplier design [11] respectively. Similarly, the throughputs of the proposed parallel and serial vector \( GF((2^8)^2) \) multipliers are 76.7% and 60.5% greater than hybrid multiplier design [10] respectively. Fig. 6 and 7 show the chip layout diagram for proposed parallel versatile and vector \( GF((2^8)^2) \) multipliers using 45 nm CMOS technology.

<table>
<thead>
<tr>
<th>( GF((2^m)^2) ) multipliers</th>
<th>MR</th>
<th>MX</th>
<th>MC</th>
<th>( x )</th>
<th>Critical path depth</th>
<th>Versatility</th>
<th>Parallel multipliers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional</td>
<td>6</td>
<td>1</td>
<td>N</td>
<td>m</td>
<td>2T(MUL)+2T(XOR)</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Hybrid [10]</td>
<td>4</td>
<td>1</td>
<td>2N</td>
<td>m</td>
<td>T(MUL)+2T(XOR)</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Karatsuba based [11]</td>
<td>4</td>
<td>1</td>
<td>N</td>
<td>m</td>
<td>2T(MUL)+T(XOR)</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Proposed parallel versatile</td>
<td>6</td>
<td>1</td>
<td>N</td>
<td>{2, 3, 4,...}m</td>
<td>T(MUL)+T(CON) +2T(XOR)</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Proposed parallel vector</td>
<td>6</td>
<td>2^x \frac{k}{2^x}</td>
<td>{2, 3, 4,...}m</td>
<td>T(MUL)+T(CON) +2T(XOR)</td>
<td>MARGINAL-INAL</td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>Proposed serial versatile</td>
<td>4</td>
<td>1</td>
<td>2N</td>
<td>{2, 3, 4,...}m</td>
<td>T(MUL)+T(CON) +2T(XOR)</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Proposed serial vector</td>
<td>4</td>
<td>2^x \frac{k}{2^x}</td>
<td>{2, 3, 4,...}m</td>
<td>T(MUL)+T(CON) +2T(XOR)</td>
<td>MARGINAL-INAL</td>
<td>YES</td>
<td></td>
</tr>
</tbody>
</table>

\( T(XOR), T(MUL) \), and \( T(CON) \) are circuit depth of 2-bit XOR gate, proposed/existing \( GF(2^m) \) multiplier, and proposed interconnect circuit respectively. \( MR \) is the number of \( GF(2^m) \) multipliers required. \( MX \) is the number of \( GF((2^2)^2) \) multiplications in parallel. \( MC \) is the number of cycles for \( N \ GF((2^2)^2) \) multiplications. Here, the conventional design follows the equations (13) and (14).
Table 4: Performance analysis for various $GF((2^8)^2)$ multiplier designs using 45 nm CMOS technology with Cadence

<table>
<thead>
<tr>
<th>$GF((2^m)^2)$ multipliers</th>
<th>Delay (ps)</th>
<th>Frequency (MHz)</th>
<th># cycles</th>
<th>Total delay (ps)</th>
<th>Throughput ($MSamples/s$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional</td>
<td>1048.6</td>
<td>953.65</td>
<td>1</td>
<td>1048.6</td>
<td>$\frac{1048.6}{1048.6 \text{ps}} = 953.74$</td>
</tr>
<tr>
<td>Hybrid [10]</td>
<td>587.1</td>
<td>1703.28</td>
<td>2</td>
<td>1174.2</td>
<td>$\frac{1174.2}{1174.2 \text{ps}} = 851.64$</td>
</tr>
<tr>
<td>Karatsuba based [11]</td>
<td>998.9</td>
<td>1001.10</td>
<td>1</td>
<td>998.9</td>
<td>$\frac{998.9}{998.9 \text{ps}} = 001.1$</td>
</tr>
<tr>
<td>Proposed parallel versatile</td>
<td>997.0</td>
<td>1003.01</td>
<td>1</td>
<td>997.0</td>
<td>$\frac{997.0}{997.0 \text{ps}} = 003.0$</td>
</tr>
<tr>
<td>Proposed parallel vector</td>
<td>1091.2</td>
<td>916.42</td>
<td>1</td>
<td>1091.2</td>
<td>$\frac{1091.2}{1091.2 \text{ps}} = 3665.6$</td>
</tr>
<tr>
<td>Proposed serial</td>
<td>914.9</td>
<td>1093.02</td>
<td>2</td>
<td>1829.8</td>
<td>$\frac{1829.8}{1829.8 \text{ps}} = 546.51$</td>
</tr>
<tr>
<td>Proposed serial vector</td>
<td>926.5</td>
<td>1079.33</td>
<td>2</td>
<td>1853.0</td>
<td>$\frac{1853.0}{1853.0 \text{ps}} = 2158.66$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Total area ($\mu m^2$)</th>
<th>Switching power (nw)</th>
<th>Leakage power (nw)</th>
<th>PDP ($\times 10^3 fJ$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional</td>
<td>2337.91</td>
<td>201275.78</td>
<td>147.32</td>
<td>211.21</td>
</tr>
<tr>
<td>Hybrid [10]</td>
<td>1748.30</td>
<td>102620.35</td>
<td>110.50</td>
<td>60.31</td>
</tr>
<tr>
<td>Karatsuba based [11]</td>
<td>1606.04</td>
<td>140450.31</td>
<td>100.72</td>
<td>140.39</td>
</tr>
<tr>
<td>Proposed parallel versatile</td>
<td>4122.84</td>
<td>95327.17</td>
<td>247.80</td>
<td>95.28</td>
</tr>
<tr>
<td>Proposed parallel vector</td>
<td>4631.01</td>
<td>167543.26</td>
<td>264.57</td>
<td>183.11</td>
</tr>
<tr>
<td>Proposed serial versatile</td>
<td>2521.20</td>
<td>75201.71</td>
<td>153.75</td>
<td>68.94</td>
</tr>
<tr>
<td>Proposed serial vector</td>
<td>2831.76</td>
<td>121057.41</td>
<td>163.39</td>
<td>112.31</td>
</tr>
</tbody>
</table>

$Power \text{ delay product (PDP) = (Switching power + Leakage power) \times Delay};  Total \text{ delay = Number of cycles} \times Delay;$  The proposed versatile $GF((2^8)^2)$ multiplier can be used to perform $GF((2^8)^2)$ multiplications, where $y \in \{2, 3, 4, 5, 6, 7, 8\}$; The proposed vector $GF((2^8)^2)$ multiplier can be used to perform one $GF((2^8)^2)$ or two $GF((2^4)^2)$ or four $GF((2^2)^2)$ or one $GF((2^4)^2)$ and two $GF((2^2)^2)$ multiplications in parallel. The conventional and existing multipliers [10] [11] can be used to perform one $GF((2^8)^2)$ multiplication. Throughput is maximum number of input or output samples per second [12].

Figure 6: Chip layout diagram for proposed parallel versatile $GF((2^8)^2)$ multiplier with core area as 4535.12 $\mu m^2$, die space around core as 2.5 $\mu m$, and total chip area as 5233.08 $\mu m^2$ using 45 nm technology.

4 Conclusion

In this paper, composite versatile and vector $GF((2^m)^2)$ multipliers are proposed. The proposed versatile $GF((2^m)^2)$ multiplier design is used to perform the $GF((2^8)^2)$ multiplication, where $2 \leq
Figure 7: Chip layout diagram for proposed parallel vector $GF((2^8)^2)$ multiplier with core area as 5094.11 $\mu m^2$, die space around core as 2.5 $\mu m$, and total chip area as 5831.84 $\mu m^2$ using 45 nm technology.

$x \leq m$. The proposed vector $GF((2^m)^2)$ multiplier design is used to perform $2^k$ numbers of $GF((2^{\frac{m}{2}})^2)$ multiplications in parallel, where throughput is comparatively higher than other designs and $k \in \{0, 1, ...,(log_2m) − 1\}$. In both the works, the hardware cost is the trade-off while the flexibility is high. The proposed and existing multipliers are synthesised and compared using 45 nm CMOS technology. The throughputs of the proposed parallel and serial vector $GF((2^8)^2)$ multipliers are 72.7% and 53.62% greater than Karatsuba based multiplier design [11] respectively.

References


